Once again, as was the case with the VLX controller, a single configurable software module is added to the existing MXE software to create the necessary functionality. As the CSA controller will, for the most part, contain (in mathematical terms), "a union of subsets" of both the LNX and VLX controllers, it will be especially easy to implement in stand-alone systems. There will, of course, be at least one new function needed for allocating channels.

This functionality includes the following features. Based on a "request for service," the central nodes selects the correct back-end machine. Once the back-end machine has been selected, i.e., a controller has been selected, it selects an available channel, i.e., a facility/timeslot. The central nodes then registers this "outgoing" triplet (i.e., the controller, the facility, and the timeslot) with the incoming controller, and registers the "incoming" triplet with the outgoing controller. Subsequent messages with these circuit id's can then be quickly routed to their respective I/O ports. (It is assumed that the necessary input/output buffering is implemented to avoid data deadlocks.) Then, the "cross-connection" between the incoming and outgoing controllers will be cleared, based on the final "release" message.

The following is a typical call scenario, using both the VLX and CSA software modules.

A call comes into a front-end machine on channel **7, 3**. The service on that machine switches the call to the central node using a "virtual" channel, say **14,9**. The signaling messages are sent over the leased X.25 line (this assumes that the VLX controller has the ability to send messages over X.25). At the central node, the VLX message will be handled by a CSA controller. The CSA controller, seeing that this is a "request for service," will first select the correct back-end machine by selecting the CSA controller for that machine (this will be based on the address information contained within the message), and then select an available channel for that CSA controller, say **3, 5**. The circuit information contained within the message will then be modified, and the message will then be sent to the out going controller. A two-dimensional array for each CSA controller will be used to maintain the "virtual" cross-connection, and the entry for each channel element will contain the triplet referencing the corresponding CSA timeslot. For subsequent messages, both messages originating from the front-end, as well as messages originating from the back-end, the CSA controller that handles the message will get the channel translation from it's matrix, (i.e., the two-dimensional array), modify the circuit information, and then send the message to the corresponding controller. This exchange will continue until the connection is cleared, as will be indicated by a "release" message.

So to summarize the changes in terms of both the service and the configuration, the multiple cluster example embodiment:

a. the number of back-end signaling links are doubled, while splitting the number of channels per link in half,

b. a second LNX controller is added on each machine, where the second front-end controller has a second link going to a standby matrix,

c. the front-end service is modified to set a prefix, and

d. the back-end service is modified to read the prefix and select an MMI hunt group.

While the invention has been described in connection with what is presently considered to be the most practical and preferred embodiments, it is to be understood that the invention is not to be limited to the disclosed embodiments, but on the contrary, is intended to cover various modifications and equivalent arrangements included within the spirit and scope of the appended claims.

What is claimed is:

1. A multimedia messaging cluster in communication with a telecommunications network, comprising:

a network exchange for communicating multimedia messages from the multimedia cluster to the telecommunications network via real network channels;

a front-end multimedia messaging exchange for communicating channel cross-connect signals to the network exchange;

at least one back-end multimedia messaging exchange communicating with the network exchange to provide the multimedia messages to the network exchange on virtual channels identified to the back-end multimedia messaging exchange by the front-end multimedia messaging exchange, wherein:

the channel cross-connect signals provided to the network exchange include cross-connect information for the network exchange to cross the multimedia messages from the virtual channels to the real network channels.

2. A cluster according to claim **1**, wherein the front-end multimedia messaging exchange includes virtual line exchange modules in a one-to-one correspondence with the back-end multimedia message exchanges, said virtual line exchange modules for communicating virtual channel switching signals to the corresponding back-end multimedia messaging exchanges.

3. A cluster according to claim **2**, wherein the virtual line exchange module communicates with a corresponding network circuit switching module located in the corresponding back-end multimedia exchanges.

4. A cluster according to claim **3**, wherein the network circuit switching module operates under the same software image whether in the cluster or in a stand-alone mode of the back-end multimedia messaging exchange.

5. A cluster according to claim **4**, wherein the virtual channel switching signals also carry subscriber list information from the back-end multimedia messaging exchanges to the corresponding virtual line exchange modules.

6. A cluster according to claim **3** wherein each pair of virtual line exchange module and corresponding network circuit switching module are assigned a predetermined group of said virtual channels.

7. A cluster according to claim **6**, wherein any virtual line exchange can assign virtual channels to a corresponding network circuit switching module only from its assigned group of virtual channels.

8. A cluster according to claim **5**, wherein the front-end multimedia messaging exchange specifies the channel cross-over signals for the respective subscribers identified by the subscriber list information.

9. A cluster according to claim **3**, wherein each network circuit switching module is a standard multimedia channel interface software module employed in a stand-alone operation of the corresponding back-end multimedia messaging exchange.

10. A cluster according to claim **1**, wherein the back-end exchange provides the multimedia messages to the network exchange via standard multimedia interfaces.

11. A cluster according to claim **5**, wherein the front-end multimedia messaging exchange further includes a controller for establishing hunt groups for forwarding messages to appropriate ones of the back-end multimedia messaging exchanges based on the subscriber list information.

12. A cluster according to claim **1**, wherein the at least one back-end multimedia messaging exchange communicates the multimedia messages to the network exchange over